

TURING DEGREES OF REALS OF POSITIVE EFFECTIVE PACKING DIMENSION

ROD DOWNEY AND NOAM GREENBERG

ABSTRACT. A relatively longstanding question in algorithmic randomness is Jan Reimann’s question whether there is a Turing cone of broken dimension. That is, is there a real A such that $\{B : B \leq_T A\}$ contains no 1-random real, yet contains elements of nonzero effective Hausdorff Dimension? We show that the answer is affirmative if Hausdorff dimension is replaced by its inner analogue packing dimension. We construct a minimal degree of effective packing dimension 1.

This leads us to examine the Turing degrees of reals with positive effective packing dimension. Unlike effective Hausdorff dimension, this is a notion of complexity which is shared by both random and sufficiently generic reals. We provide a characterization of the c.e. array noncomputable degrees in terms of effective packing dimension.

1. INTRODUCTION

This paper is concerned with effective notions of randomness and their relationship with measures of computational complexity such as Turing degrees. This programme has been around since the 1950’s beginning with the work of de Leeuw, Moore, Shannon and Shapiro [4]. A new initiative in the study of algorithmic randomness was the work of Staiger, Lutz and his co-authors, and others (e.g. [1, 16, 17, 24]) who effectivized the refinements of the notion of Lebesgue measure known as dimensions. The best known of these is the notion of Hausdorff dimension [11]. For our purposes, we will take as our definition for this notion a characterization due to Mayordomo [18] which is that the effective Hausdorff dimension of a real $A \in 2^\omega$ is

$$\dim(A) = \liminf_{n \rightarrow \infty} \frac{K(A \upharpoonright n)}{n}.$$

Here K denotes prefix-free Kolmogorov complexity (though plain complexity C would be okay as well) and $A \upharpoonright n$ denotes the first n bits of the real A . The easiest way to construct a real of fractional effective dimension is to “water down” a random real. For example, if B is Martin-Löf random then by Schnorr’s theorem, for all n , $K(B \upharpoonright n) \geq n - c$ for some fixed constant c . Then the real obtained by inserting a 0 between any two bits of B will have effective Hausdorff dimension $\frac{1}{2}$.

It is a longstanding question whether in some sense this is the *only* way to construct a real of broken dimension: to take a random real and water it down in some effective way. If this were the only way, then the process of watering down could be effectively reversed, which would mean that we could extract random information from every real of positive effective Hausdorff dimension. This question was first articulated by Jan Reimann in his thesis [21] :

The first author’s research was supported by the Marsden Fund of New Zealand.

If A has positive effective Hausdorff dimension, is there a random real $B \leq_T A$?

The reader should also see Downey, Hirschfeldt, Nies and Terwijn [6] and Miller and Nies [19].

There are of course many other notions of dimension, and hence of effective dimension, in the study of measure and effective measure. The broken dimension question has been open for all such dimensions. Here we solve the question for the notion of effective packing dimension, which is an inner measure version, based around packing with balls of shrinking radius, of the notion of effective dimension, effectivizing the definition from, say, Falconer [9]. Again we will use a characterization as our working definition of effective packing dimension. This is a characterization due to Lutz [17]: the effective packing dimension of a real $A \in 2^\omega$ is

$$\text{Dim}(A) = \limsup_{n \rightarrow \infty} \frac{K(A \upharpoonright n)}{n} = \limsup_{n \rightarrow \infty} \frac{C(A \upharpoonright n)}{n}.$$

Here we prove:

Theorem 1.1. *There is a real of minimal Turing degree and of effective packing dimension 1.*

We get the following corollary.

Corollary 1.2. *There is a real A of effective packing dimension 1, such that for all reals $B \leq_T A$, B is not random.*

The corollary follows since it is well known that no random real can have minimal Turing degree. (See e.g. Downey and Hirschfeldt [5] or Nies [20].)

Theorem 1.1 raises a more general question: what kind of Turing degrees contain reals with positive effective packing dimension? we note that having effective packing dimension one is a property that is shared by both random reals and sufficiently generic reals (so the class of real that have positive effective packing dimension has both full measure and is co-meagre; in contrast, the class of reals that have positive effective Hausdorff measure has measure 1 but is meagre.) This is why our gaze turns naturally to the class of array noncomputable degrees, introduced by Downey, Jockusch and Stob [7, 8]. In the case of c.e. degrees, we get a complete characterisation.

Recall that a Turing degree \mathbf{a} is *array noncomputable* if for all $f \leq_{wt} \emptyset'$, there is a function $g \leq_T \mathbf{a}$ such that $\exists^\infty n (g(n) > f(n))$. The array noncomputable degrees were introduced to explain a number of “multiple permitting” arguments in computability theory. Since then we have realized that this is an important class capturing a wide collection of apparently unrelated phenomena.

Early on, it was realized that at least in the c.e. case, array noncomputability is intimately related to Kolmogorov complexity. An old result of Barzdins' [2] states that if A is a c.e. set then $C(A \upharpoonright n) \leq 2 \log n + O(1)$. In unpublished work Solovay (see [5]) showed that it is not possible for a c.e. set to have $C(A \upharpoonright n) \geq 2 \log n + O(1)$ for all n . It was open whether there were “complex” c.e. sets in that $C(A \upharpoonright n) \geq 2 \log n + O(1)$ for *infinitely many* n . Kummer proved the following gap theorem. Recall that an *order function* is an unbounded and nondecreasing computable function.

Theorem 1.3 (Kummer’s Gap Theorem [13]).

(i) *A c.e., array noncomputable degree contains a c.e. set A such that*

$$C(A \upharpoonright n) \geq 2 \log n - O(1)$$

for infinitely many n .

(ii) *If A has c.e., array computable degree, then for every order function h ,*

$$C(A \upharpoonright n) \leq \log n + h(n) + O(1).$$

Thus for no $\varepsilon > 0$ do we have

$$C(A \upharpoonright n) \geq (1 + \varepsilon) \log n - O(1)$$

for infinitely many n .

Note that part (ii) holds for all elements of the c.e. degree, not only the c.e. ones. It implies that a c.e., array computable degree cannot contain any real with positive effective packing dimension. Kummer's proof, in fact, can be easily generalised. Recall that a degree \mathbf{a} is *c.e. traceable* if there is some order h such that for all $f \leq_T \mathbf{a}$ there is a uniformly c.e. sequence $\langle T_x \rangle$ such that for all x , $|T_x| \leq h(x)$ and $f(x) \in T_x$. All c.e. traceable degrees are array computable. C.e. traceability was introduced by Zambella [26] and independently by Ishmukhametov [12], who noted that in the c.e. degrees, c.e. traceability and array computability coincided. Hence the second part of Kummer's theorem follows from:

Proposition 1.4. *If A is c.e. traceable, then for every order function h ,*

$$C(A \upharpoonright n) \leq \log n + h(n) + O(1).$$

Sketch of proof. This is similar to Kummer's proof. We know that if A is c.e. traceable then any order function h can serve for a bound for the size of traces of functions computable from h (see Terwijn and Zambella [25].) We trace the function $n \mapsto A \upharpoonright n$. Let $\langle T_n \rangle$ serve as a trace for this function with bound h . Then to specify $A \upharpoonright n$ we need n (this takes $\log n$ many bits), a constant number of bits to describe the machine enumerating $\langle T_n \rangle$, and $\log h(n)$ many bits to tell which member of T_n (in order of enumeration) $A \upharpoonright n$ is. \square

Thus no c.e. traceable degree can contain a real whose effective packing dimension is positive. We note that this, together with Theorem 1.1, implies the fact that there is a minimal degree which is not c.e. traceable (see for example [10]). This shows that c.e. traceability is not sufficient to settle Yates' question on strong minimal covers of minimal degrees (Ishmukhametov [12] showed that every c.e. traceable degree has a strong minimal cover.) We further mention that adding the priority method to a partial-tree version of our notion of forcing would produce a minimal degree below $\mathbf{0}'$ which has an element of effective packing dimension 1. This implies Gabbay's result [10] that there is a minimal degree below $\mathbf{0}'$ which is not c.e. traceable, thus showing that Ishmukhametov's partition of the c.e. degrees does not extend to the Δ_2^0 degrees.

On the positive side, we have the following analogue of (i) of Kummer's theorem:

Theorem 1.5. *Every array noncomputable degree \mathbf{a} computes a set A with effective packing dimension 1. If \mathbf{a} is a c.e. degree then A can be taken to be a left-c.e. real, even a left-c.e. real which is a point of rank 1 in a countable Π_1^0 class. Thus a c.e. degree computes a real with positive effective packing dimension iff it is array noncomputable.*

Part of the theorem is an elaboration on a recent result by Conidis [3], who constructed a countable Π_1^0 class P (indeed one whose Cantor-Bendixson rank is 1) whose effective packing dimension is 1; equivalently, the unique nonisolated path in P has effective packing dimension 1. This shows that the effective and noneffective versions of packing dimension differ on some Π_1^0 classes; whereas it is known that the effective and noneffective versions of Hausdorff dimension must agree on such classes. Our result shows that Conidis' example can be found below any c.e., array noncomputable degree.

Notation is standard and generally follows Soare [23]. For more on dimension and related notions of Kolmogorov complexity, the reader should also see Downey, Hirschfeldt, Nies and Terwijn [6], Downey and Hirschfeldt [5] or Nies [20].

2. CLUMPY TREES

We prove Theorem 1.1 by presenting a notion of forcing \mathbb{P} such that a sufficiently generic filter $G \subset \mathbb{P}$ yields a real $X_G \in 2^\omega$ which has effective packing dimension 1 and minimal Turing degree. This is a modification of the standard forcing with computable perfect trees due to Sacks [22]. We need to restrict the kind of perfect trees we use so that we can always choose strings that are sufficiently complicated (i.e., not easily compressed), to be initial segments of the real we build. The problem, of course, is that we cannot determine effectively which strings are sufficiently incompressible, but our conditions, the trees, have to be computable. The solution to this problem relies on the following lemma.

Lemma 2.1. *There is a computable mapping $(\sigma, \varepsilon) \mapsto n_\varepsilon(\sigma)$ which maps a finite binary string $\sigma \in 2^{<\omega}$ and a positive rational ε to a natural number n such that there is some binary string τ of length n such that*

$$\frac{K(\sigma\tau)}{|\sigma\tau|} \geq 1 - \varepsilon.$$

Proof. Let $d = |\sigma| + 1$. We know that the measure of the set of strings ν such that $K(\nu) \leq |\nu| - d$ is at most 2^{-d} , and so cannot contain $[\sigma]$. Let $m > d/\varepsilon$. Then there is some ν extending σ of length m such that $K(\nu) \geq m - d$. Then

$$\frac{K(\nu)}{m} \geq 1 - \frac{d}{m} > 1 - \varepsilon.$$

We can thus let $n = m - |\sigma|$. □

The idea is to ensure that the trees we use are “clumpy”. Suppose that a string σ lies on a tree T . If every extension of σ of length $|\sigma| + n_\varepsilon(\sigma)$ also lies on T , then we could (non effectively) choose one of those extensions which is incompressible. If we refine the tree and remove one of these extensions, then we commit to removing σ as well. A clumpy tree is one which has cofinally many (larger and larger) clumps.

Recall that a *perfect function tree* is a map $T: 2^{<\omega} \rightarrow 2^{<\omega}$ which preserves inclusion and incomparability.

Let T be a perfect function tree, $\sigma \in \text{im } T$, and let ε be a positive rational. We say that T *contains an ε -clump above σ* if for all binary strings τ of length $n_\varepsilon(\sigma)$,

$$\sigma\tau \subseteq T(\rho\tau),$$

where $\rho = T^{-1}(\sigma)$.

Given a perfect function tree T and a positive rational ε , we recursively define a labeling of some of the image of T :

- (1) Label $T(\langle \rangle)$ by ε .
- (2) If σ is labeled by a rational number δ , and T contains a δ -clump above σ , then for all binary strings τ of length $n_\delta(\sigma)$, label $T(\rho\tau)$ by $\delta/2$, where $\rho = T^{-1}(\sigma)$.

If T does not contain such a clump, stop the labeling process.

A tree T is called ε -clumpy if the labeling process of T (starting with ε) never halts, that is, the δ -clumps required at step 2 are always present.

Let \mathbb{P} be the collection of pairs (T, ε) where T is an ε -clumpy, computable tree. Note that $\mathbb{P} \neq \emptyset$; for example, the full binary tree $\text{id}_{2^{<\omega}}$ is ε -clumpy for all $\varepsilon > 0$.

Recall that we say that a perfect tree S *extends* a perfect tree T if there is some perfect tree R such that $S = T \circ R$. If $(T, \varepsilon), (S, \delta) \in \mathbb{P}$ then we say that (S, δ) *extends* (T, ε) if S extends T and δ is the label of $S(\langle \rangle)$ on (T, ε) (in particular, we do require that $S(\langle \rangle)$ is labeled on (T, ε) .)

A standard example is that of *full* subtrees: If T is a perfect tree and $\sigma \in \text{im } T$, then

$$\mathbf{Ext}_\sigma(T) = T \circ \{\tau \mapsto \rho\tau\},$$

where $\rho = T^{-1}(\sigma)$ is the full subtree of T issuing from σ . If $(T, \varepsilon) \in \mathbb{P}$ and $\sigma \in \text{im } T$ is labeled by δ , then $(\mathbf{Ext}_\sigma(T), \delta)$ is in \mathbb{P} and extends (T, ε) .

If $G \subset \mathbb{P}$, we let $X_G = \bigcap_{(T, \varepsilon) \in G} [T]$ (as usual, $[T] = \{X \in 2^\omega : \exists^\infty n X \upharpoonright n \in \text{im } T\}$.)

Lemma 2.2. *If $G \subset \mathbb{P}$ is a sufficiently generic filter, then X_G is an infinite binary string, and $\text{Dim}(X_G) = 1$ (in particular, X_G is not computable.)*

Proof. By taking full subtrees, we can ensure that $T(\langle \rangle)$ is as long as we like for some $(T, \varepsilon) \in G$.

Let $q < 1$ and let $(T, \varepsilon) \in \mathbb{P}$. There is some ρ such that on (T, ε) , $\sigma = T(\rho)$ is labeled by δ where $1 - \delta > q$. Let $n = n_\delta(\sigma)$. There is some τ of length n such that $K(\sigma\tau)/|\sigma\tau| > 1 - \delta > q$. Then $T(\rho\tau)$ is labeled by $\delta/2$ on (T, ε) and $(\mathbf{Ext}_{T(\rho\tau)}(T), \delta/2) \in \mathbb{P}$ extends (T, ε) . If that condition is in G then $\sigma\tau \subseteq T(\rho\tau) \subset X_G$. Thus the set of conditions that forces that there is some initial segment ν of X_G such that $K(\nu)/|\nu| > q$ is dense in \mathbb{P} . \square

Next we show that if G is sufficiently generic then X_G has minimal Turing degree. Suppose that Φ is a Turing functional. As usual in Sacks forcing, let

$$\mathbf{Div}_\Phi = \{(T, \varepsilon) \in \mathbb{P} : \exists x \forall \sigma \in \text{im } T (\Phi^\sigma(x) \uparrow)\}$$

and let

$$\mathbf{Tot}_\Phi = \{(T, \varepsilon) \in \mathbb{P} : \forall x \forall \sigma \in \text{im } T \exists \sigma' \in \text{im } T (\sigma' \supseteq \sigma \ \& \ \Phi^{\sigma'}(x) \downarrow)\}.$$

The standard argument shows that $\mathbf{Tot}_\Phi \cup \mathbf{Div}_\Phi$ is dense in \mathbb{P} .

Let \mathbf{Comp}_Φ be the collection of conditions $(T, \varepsilon) \in \mathbf{Tot}_\Phi$ such that for all $\sigma, \sigma' \in \text{im } T$, Φ^σ and $\Phi^{\sigma'}$ are comparable. A standard argument shows that if $(T, \varepsilon) \in \mathbf{Comp}_\Phi$ and $(T, \varepsilon) \in G$ then $\Phi(X_G)$ is computable.

A condition $(T, \varepsilon) \in \mathbf{Tot}_\Phi$ is called *clumpily- Φ -splitting* if for all incomparable, labeled $\sigma, \sigma' \in \text{im } T$, Φ^σ and $\Phi^{\sigma'}$ are incomparable. The standard argument from Sacks forcing shows that if $(T, \varepsilon) \in \mathbf{Tot}_\Phi$ is clumpily- Φ -splitting then $X_G \leq_T \Phi(X_G)$. This is because the labeled nodes are dense in $\text{im } T$. Our proof ends, therefore, with the following lemma:

Lemma 2.3. *Let \mathbf{Sp}_Φ be the collection of clumpily- Φ -splitting conditions. Then $\mathbf{Sp}_\Phi \cup \mathbf{Comp}_\Phi$ is dense below \mathbf{Tot}_Φ in \mathbb{P} .*

Proof. Suppose that $(T, \varepsilon) \in \mathbf{Tot}_\Phi$ and has no extension in \mathbf{Comp}_Φ . This means that for all $\sigma \in \text{im} T$, there are $\sigma_0, \sigma_1 \in \text{im} T$ extending σ such that Φ^{σ_0} and Φ^{σ_1} are incomparable.

By recursion, we define an extension (S, ε) of (T, ε) in \mathbf{Sp}_Φ . We start with letting $S(\langle \rangle) = T(\langle \rangle)$ and label this string by ε on (S, ε) as well. Suppose that we defined $\sigma = S(\rho') = T(\rho)$ where σ is labeled by δ on (T, ε) and by some γ on (S, ε) ; we ensure that $n_\gamma(\sigma) \leq n_\delta(\sigma)$. For every τ of length strictly shorter than $n_\gamma(\sigma)$, we can let $S(\rho'\tau) = T(\rho\tau) = \sigma\tau$ (and leave it unlabeled on (S, ε)).

Let $\tau_1, \dots, \tau_{2^n}$ be all the strings of length $n = n_\gamma(\sigma)$. Enumerate the pairs of numbers (i, j) such that $1 \leq i < j \leq 2^n$ as (i_k, j_k) for $k \leq k^*$. For each $i \leq 2^n$ and $k \leq k^*$ define strings ν_k^i such that $\sigma\tau_i \subset \nu_0^i \subset \dots \subset \nu_{k^*}^i$ and such that every ν_k^i is in the image of T . At step k , define $\nu_k^{i_k} \supset \nu_{k-1}^{i_k}$ and $\nu_k^{j_k} \supset \nu_{k-1}^{j_k}$ such that $\Phi^{\nu_k^{i_k}} \perp \Phi^{\nu_k^{j_k}}$; this is possible because (T, ε) has no extension in \mathbf{Comp}_Φ . We then define $S(\rho'\tau_i)$ be some string $\varsigma_i \in \text{im} T$ extending $\nu_{k^*}^i$ which is labeled by some δ_i such that $n_{\delta_i}(\varsigma_i) \geq n_{\gamma/2}(\varsigma_i)$. We can then label ς_i by $\gamma/2$ on (S, ε) . \square

3. ARRAY COMPUTABILITY AND EFFECTIVE PACKING DIMENSION

We turn to prove Theorem 1.5. In the most general case, recall from [8] that a function f is *pb-approximable* if there is some computable approximation $f(x, s)$ of f and some primitive recursive bound h for the “mind-change” function $x \mapsto \#\{s : f(x, s) \neq f(x, s+1)\}$. A dense set $X \subset 2^{<\omega}$ is called *pb-dense* if there is a pb-approximable function $f: 2^{<\omega} \rightarrow 2^{<\omega}$ such that for all $\sigma \in 2^{<\omega}$, $\sigma \subseteq f(\sigma)$ and $f(\sigma) \in X$. A real $A \in 2^\omega$ is *pb-generic* if it meets every pb-dense set of strings. Downey, Jockusch and Stob showed that every array noncomputable degree bounds a pb-generic real.

Proposition 3.1. *If A is pb-generic then $\text{Dim}(A) = 1$.*

Proof. This is because the map $(\sigma, k) \mapsto n_{1/k}(\sigma)$ given by Lemma 2.1 is polynomial (in $|\sigma|$ and k), and so $(\sigma, k) \mapsto 2^{n_{1/k}(\sigma)}$ is primitive recursive. For $k > 0$, let

$$X_k = \{\nu \in 2^{<\omega} : |\nu| > k \ \& \ K(\nu)/|\nu| > 1 - 1/k\}.$$

Note that X_k is Π_1^0 (co-c.e.) Also, X_k is pb-dense, because we can approximate some $f(\sigma) \supseteq \sigma$ in X_k by choosing an extension ν of length $|\sigma| + n_{1/k}(\sigma)$ and abandoning it if it is later discovered that ν leaves X_k , choosing another such extension instead, until one is chosen that never leaves X_k . \square

Now suppose that D is a c.e. set which has array noncomputable degree. To obtain multiple permitting from D , we use the fact [7] that we may assume that if $\langle I_m \rangle$ is a computable partition of ω into rapidly increasing intervals (the size of the intervals does not matter, as long as it increases with m), then for every c.e. set W there are infinitely many m such that $W \cap I_m = D \cap I_m$. Thus to get permissions to change some $A \leq_T D$ that we are building, we tie the use of computing $A \upharpoonright n$ from D to include one of these intervals I_m ; to get changes in D , we enumerate numbers from I_m into an auxiliary set Q (note that we may assume that at all stages, $D \cap I_m \subseteq Q \cap I_m$ or we can maintain a disagreement by freezing Q .)

To build a left-c.e. real $A \leq_T D$ which has effective packing dimension 1, we meet the requirements

R_e : There is some $\nu \subset A$ of length at least e such that $K(\nu)/|\nu| > 1 - 1/e$.

That is, we want some $\nu \subset A$ in X_e (from the proof of Proposition 3.1).

The strategy for meeting the first requirement (say R_2 , because R_0 is meaningless and R_1 is automatic) is the following. We know that $\langle \rangle \subset A$; we want to direct A to extend some extension of $\langle \rangle$ in X_2 . Let $n = n_{1/2}(\langle \rangle)$ (so we know that there is some string of length n in X_2) and find some m such that $|I_m| > 2^n$. Direct A to extend the leftmost string ν of length n which currently looks like it is in X_2 . Set the use of computing $A \upharpoonright n = \nu$ from D to be $\max I_m + 1$. The requirement is currently satisfied.

Suppose that at some stage s , we discover that ν left X_2 . We then seek permission from D to change $A \upharpoonright n$ by enumerating $x = \min I_m$ into an auxiliary set Q_2 . We wait for x to appear in D , thus changing D below the use of computing $A \upharpoonright n$ and allowing us to move to the right, to the currently left-most string of length n in X_2 (note again that X_2 is co-c.e., so no string that lies to the left of ν can re-appear in X_2 ; so we never move to the left, only to the right, so A is indeed left-c.e.) While we wait for the permission to occur, we think that this attempt has failed (because we picked the wrong interval I_m) and so we retry higher up: we assume that $\nu = \nu_0$ is an initial segment of A , so we calculate $n_1 = |\nu_0| + n_{1/2}(\nu_0)$ and direct A to pass through the leftmost extension ν_1 of ν_0 of length n_1 which is currently in X_2 . We pick some large m_1 , set the use of calculating $A \upharpoonright n_1$ from D to be $\max I_{m_1} + 1$, and repeat the process. If we later receive permission to change $A \upharpoonright n_0$ (where $n_0 = n = |\nu_0|$) then we abandon all parameters n_k, m_k, ν_k for $k > 0$ and repeat the process with the new initial segment of A of length n_0 .

In the event that we get a stage at which $D \cap I_{m_k} \not\subseteq Q_2 \cap I_{m_k}$ (because of some enumeration into $D \cap I_{m_k}$), we abandon m_k and redefine it to be some large m ; the D -change enables us to change the use of calculating $A \upharpoonright n_k$ beyond the maximum of the new I_m .

We then argue by induction that at every level n_k , $A \upharpoonright n_k$ eventually stabilizes, and so the parameter n_{k+1} stabilizes as well. Also, m_{k+1} stabilizes because we cannot always have $D \cap I_{m_k} \not\subseteq Q_2 \cap I_{m_k}$. Thus on some permanent m_k we indeed get $D \cap I_{m_k} = Q_2 \cap I_{m_k}$ and so we get sufficiently many permissions and meet the requirement R_2 .

Thus the action of R_2 is finitary and the whole construction is a finite injury one. This makes the interaction of R_2 with weaker requirements easy: each time R_2 acts, all weaker requirements are initialised (and all of their own parameters n_k, m_k, ν_k are forgotten). The only thing that we need to note is the fact that the reduction of A to D is global, and so R_2 has to take into consideration the ‘‘litter’’ left by initialised weaker requirements, namely, reductions of some initial segments of A to D , which we may assume are permanent. This is of no real difficulty. For if R_2 is currently directing A to pass via $\nu_1 \subset \nu_2 \subset \dots \subset \nu_l$ (each ν_k of length n_k and the use of $A \upharpoonright n_k$ is $\max I_{m_k} + 1$), then the definitions made by weaker requirements are all for strings extending ν_l . If R_2 receives permission to change some ν_k then the D -change enables us to forget about all such promises made by weaker requirements. If the R_2 action is because of some failure to receive permission, then R_2 is trying to define n_{l+1}, m_{l+1} and ν_{l+1} . Then, instead of taking the new base to be ν_l (and defining $n_{l+1} = n_l + n_{1/2}(\nu_l)$), we let $\tau = A \upharpoonright s[s]$ be the longest initial segment of

A for which some promises were made by anyone in the construction, promise that if $\nu_l \subset A$ then $\tau \subset A$, and let $n_{l+1} = |\tau| + n_{1/2}(\tau)$, and try to meet R_2 by finding extensions of τ in X_2 .

Note that in fact we get $A \leq_{wtt} D$.

Now to get A to be the unique rank 1 point in a Π_1^0 class P of rank 1, we change the construction as follows. We start with $P = 2^{<\omega}$. At stage s , if some requirement R_e is defining $\nu_k \subset A$ because $n_k = |\nu_k| = |\tau_k| + n_{1/e}(\tau_k)$ (where it is agreed that if $\nu_{k-1} \subset A$ then $\tau_k \subset A$) then we leave not only ν_k on $P[s]$ but also every other extension of τ_k of length n_k . That is, the entire $1/e$ -clump above τ_k is left in P . But apart of the clumps, we remove all strings of length s except for those that are of the form $\sigma 0^l$ for some σ in some such clump. Upon redefining some n_k and ν_k , we need of course to pick some τ_k extending ν_{k-1} which is still on P . If at stage s we pick one of length s , then the entire clump above τ_k is still present in P and we can protect it.

In the end, A will still be the unique nonisolated path in $[P]$. For if $\sigma \in P$ lies to the left of A , then after some stage, A moves to the right of σ and we stop putting clumps above σ in P ; so all splitting ceases above σ beyond some level. And if σ lies to the right of A , then we never put any clump above σ in P , so $[P] \cap [\sigma] = \{\sigma 0^\omega\}$.

REFERENCES

- [1] Artheya, K., Hitchcock, J., Lutz, J., and E. Mayordomo, *Effective strong dimension, algorithmic information, and computational complexity*, Proceedings of the Twenty-First Symposium on Theoretical Aspects of Computer Science (Montpellier, France, March 25–27, 2004), Springer-Verlag, 2004, pp. 632–643.
- [2] Barzdins, J., *Complexity of programs to determine whether natural numbers not greater than n belong to a recursively enumerable set*, Soviet Mathematics Doklady vol. 9 (1968), 1251–1254.
- [3] Conidis, C., *PhD Thesis*, University of Chicago, in preparation.
- [4] de Leeuw, K., Moore, E. F., Shannon, C. E., and Shapiro, N., *Computability by probabilistic machines*, Automata studies, pp. 183–212. Annals of mathematics studies, no. 34. Princeton University Press, Princeton, N. J., 1956.
- [5] Downey, R. and Hirschfeldt, D., *Algorithmic Randomness and Complexity*, Springer-Verlag, to appear.
- [6] Downey, R., Hirschfeldt, D., Nies, A. and Terwijn, S., *Calibrating randomness*, Bulletin of Symbolic Logic, Vol. 3 (2006), 411–491.
- [7] Downey, R., C. Jockusch, and M. Stob, *Array nonrecursive sets and multiple permitting arguments*, in *Recursion Theory Week* (Ambos-Spies, Muller, Sacks, eds.) Lecture Notes in Mathematics 1432, Springer-Verlag, Heidelberg, 1990, 141–174.
- [8] Downey, R., C. Jockusch, and M. Stob, *Array nonrecursive degrees and genericity*, in *Computability, Enumerability, Unsolvability* (Cooper, Slaman, Wainer, eds.), London Mathematical Society Lecture Notes Series 224, Cambridge University Press (1996), 93–105.
- [9] Falconer, K., *Fractal geometry, mathematical foundations & applications*, Wiley & Sons, 1992.
- [10] Gabbay, Y., *Double Jump Inversions and Strong Minimal Covers in the Turing Degrees*, PhD Thesis, Cornell University, 2004.
- [11] Hausdorff, F., *Dimension und äußeres Maß*, Mathematische Annalen 79 (1919) 157–179.
- [12] Ishmukhametov, S., *Weak recursive degrees and a problem of Spector*, in *Recursion Theory and Complexity*, (ed. M. Arslanov and S. Lempp), de Gruyter, (Berlin, 1999), 81–88.
- [13] Kummer, M., *Kolmogorov complexity and instance complexity of recursively enumerable sets*, SIAM Journal of Computing, Vol. 25 (1996), 1123–1143.
- [14] Lerman, M., *Degrees of unsolvability. Local and global theory*, Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1983. xiii+307 pp.
- [15] Li, M. and Vitanyi, P., *Kolmogorov Complexity and its Applications*, Springer-Verlag, 1993.

- [16] Lutz, J., *Category and measure in complexity classes*, SIAM J. Comput. 19 (1990) 1100–1131.
- [17] Lutz, J., *Effective fractal dimensions*, Mathematical Logic Quarterly Vol. 51 (2005), pp. 62–72.
- [18] Mayordomo, E., *A Kolmogorov complexity characterization of constructive Hausdorff dimension*, Information Processing Letters 84 (2002) 1–3.
- [19] Miller J. S. and Nies, A., *Randomness and computability: open questions*, Bull. Symb. Logic. 12 no 3 (2006) 390–410.
- [20] Nies, A., *Computability and Randomness*, Oxford University Press, in preparation.
- [21] Reimann, J., *Computability and Dimension*, unpublished notes, University of Heidelberg, 2004
- [22] Sacks, G. E., *Forcing with perfect closed sets*. 1971 Axiomatic Set Theory (Proc. Sympos. Pure Math., Vol. XIII, Part I, Univ. California, Los Angeles, Calif., 1967) pp. 331–355 Amer. Math. Soc., Providence, R.I.
- [23] Soare, R., *Recursively enumerable sets and degrees*, Springer, Berlin, 1987.
- [24] Staiger, L., *Kolmogorov complexity and Hausdorff dimension*, Information and Computation, Vol. 103 (1993), 159–194.
- [25] Terwijn, S., and Zambella, D., *Algorithmic randomness and lowness*, Journal of Symbolic Logic, vol. 66 (2001), 1199–1205.
- [26] Zambella, D., *On sequences with simple initial segments*, ILLC technical report, ML-1990-05, University of Amsterdam, 1990.

SCHOOL OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, VICTORIA UNIVERSITY, P.O.
 BOX 600, WELLINGTON, NEW ZEALAND
E-mail address: `Rod.Downey@vuw.ac.nz`

SCHOOL OF MATHEMATICS, STATISTICS AND COMPUTER SCIENCE, VICTORIA UNIVERSITY, P.O.
 BOX 600, WELLINGTON, NEW ZEALAND
E-mail address: `greenberg@mcs.vuw.ac.nz`